

I. REAL PARTY IN INTEREST	1
II. RELATED APPEALS AND INTERFERENCES	1
III. STATUS OF CLAIMS.....	2
IV. STATUS OF AMENDMENTS.....	2
V. SUMMARY OF CLAIMED SUBJECT MATTER.....	2
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL.....	3
VII. ARGUMENT.....	4
VIII. CLAIMS APPENDIX	24
IX. EVIDENCE APPENDIX	30
X. RELATED PROCEEDINGS APPENDIX	31

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of	:	Customer Number: 46320
	:	
Iwao INAGAKI, et al.	:	Confirmation Number: 5257
	:	
Application No.: 10/715,769	:	Group Art Unit: 2192
	:	
Filed: November 18, 2003	:	Examiner: J. Rutten
	:	
For: MULTITHREAD TRACING METHOD AND APPARATUS	:	

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is submitted in support of the Notice of Appeal filed February 21, 2007, wherein Appellants appeal from the Examiner's rejection of claims 1-22.

I. REAL PARTY IN INTEREST

This application is assigned to IBM Corporation by assignment recorded on November 18, 2003, at Reel 014715, Frame 0230.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals and interferences.

III. STATUS OF CLAIMS

Claims 1-22 are pending and finally rejected in this Application. It is from the final rejection of claims 1-22 that this Appeal is taken.

IV. STATUS OF AMENDMENTS

The claims have not been amended subsequent to the imposition of the Second Office Action dated November 21, 2006 (hereinafter the Second Office Action).

V. SUMMARY OF CLAIMED SUBJECT MATTER

Referring to Figures 2 and 3 and also to independent claims 1 and 12, an information processor 21 and method suitable for executing a routine including a tracing process for collecting information related to multithreading processing status is disclosed. The information processor includes routine registration means and level determination means. The routine registration means registers a routine being executed for each thread being activated (lines 1-3 of paragraph [0032] of Appellants' disclosure). The level determination means determines a level of a tracing process for the routine being executed based on registration information registered by the routine registration means (lines 4-8 of paragraph [0032]). Referring to Figure 7, and to dependent claims 2 and 13, in steps 84 and 85, the level determination means performs the level determination of the tracing process for the routine being executed based on whether a routine identical to, or in a predetermined relation with, the routine is registered. (lines 1-12 of paragraph [0053]).

Referring to dependent claims 3 and 14, the information processor includes a table 22 in which routines having a predetermined mutual relation are specified (lines 1-8 of paragraph

[0031]). Referring to dependent claims 4 and 15, the information processor accepts input for setting or changing the table 22 (lines 1-10 of paragraph [0032]). Referring to dependent claims 5 and 16, the routines in a predetermined relation are routines that access a shared resource (lines 1-11 of paragraph [0043]).

Referring to dependent claims 6 and 17, the level determination means indicates permission, in determining the level of the tracing process for the routine being executed, for unrestricted execution of a tracing process if a routine identical to, or in a predetermined relation with, the routine being executed is registered (lines 5-7 of paragraph [0031]). Referring to dependent claims 7 and 18, the level determination means blocks, in determining the level of the tracing process for the routine being executed, execution of a tracing process unless a routine identical to, or in a predetermined relation with, the routine being executed is registered (lines 1-4 of paragraph [0041]).

Referring to dependent claims 8 and 19, responsive to registration of a routine, the level determination means adds, if there is already registered a routine identical to or in a predetermined relation with the routine being registered, simultaneous execution information to registration information of both the routine being registered and the routine already registered (lines 1-6 of paragraph [0045]). Referring to dependent claims 9 and 20, the simultaneous execution information includes trace level information indicating the level of the tracing process (lines 1-6 of paragraph [0045]).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-5 and 12-16 were rejected under 35 U.S.C. § 101;
2. Claims 1-4, 8-9, 12-15, and 19-20 were rejected under 35 U.S.C. § 102 for anticipation based upon Alexander et al., U.S. Patent No. 6,055,492 (hereinafter Alexander);

3. Claims 5 and 16 were rejected under 35 U.S.C. § 103 for obviousness based upon Alexander in view Broberg et al., "Visualization and Performance Prediction" (hereinafter Broberg);

4. Claims 6-7 and 17-18 were rejected under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Larus, "Efficient Program Tracing";

5. Claims 10 and 21 were rejected under 35 U.S.C. § 103 for obviousness based upon Alexander in view of "Debugging with GDB" (hereinafter GDB); and

6. Claims 11 and 22 were rejected under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Sauvage, U.S. Patent No. 6,163,881.

VII. ARGUMENT

THE REJECTION OF CLAIMS 1-5 AND 12-16 UNDER 35 U.S.C. § 101

For convenience of the Honorable Board in addressing the rejections, claims 2-5 stand or fall together with independent claim 1, and claims 13-16 stand or fall together with independent claim 12.

The subject matter to which in independent claims 1 and 12 are respectively directed, is an information processor (i.e., a device) and a method. Both a device and a method are statutory subject matter. In the thirteenth enumerated paragraph spanning pages 6 of the Second Office Action, the Examiner asserted the following:

13. Claims 1-5 and 12-16 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Claim 1 is directed to an "information processor" and recites means for "registering a routine" and "determining a level of a tracing process". However, these limitations fall short of actually producing a useful and tangible result. While the elements of the claim are necessary for providing the utility as disclosed in the specification (e.g. page 2 paragraph [008] and page 6 paragraph [030]), they do not actually provide any useful result in the form of enabling control or analysis of threads or routines that access shared resources. In contrast, claim 6 is exemplary for using an indication of "permission ... for unrestricted execution" of

identically registered routines. In this case a useful and tangible result is obtained, and the claim is statutory. Claim 12 is a method claim including similar limitations to claim 1, and is rejected for the same reasons as provided above. Claims 2-5 and 13-16 are dependent upon rejected base claims and are rejected for similar reasons as presented above. For further information, see MPEP 2106.

With certain exceptions not encompassed by the present claims (i.e., via the use of the claim language of "consisting"), claims are not exclusionary. In other words, the subject matter encompassed by the claims does not exclude subject matter not specifically recited by the claims. Thus, as argued on pages 2 and 3 of the Request for Reconsideration filed September 6, 2006 (hereinafter the Response), dependent claims 1 and 12 include all of the limitations recited in independent claims 6 and 17, respectively, and the only difference between claims 1 and 6 and claims 12 and 17 is that claim 6 has a narrower scope than claim 1 and claim 17 has a narrower scope than claim 12. Thus, if claims 6 and 17 are directed to statutory subject matter, then claims 1 and 12 must also cover statutory subject matter since the subject matter covered by claims 1 and 12, by definition, respectively include the subject matter covered by claims 6 and 17.

Appellants are unaware of any requirement that the claims themselves must recite the useful, concrete, and tangible result. Instead, reference is made to M.P.E.P. § 2107.02(I), which states:

regardless of the category of invention that is claimed (e.g., product or process), an applicant need only make one credible assertion of specific utility for the claimed invention to satisfy 35 U.S.C. 101 and 35 U.S.C. 112.

Moreover, as noted in M.P.E.P. § 2107.02(III)(A), the Court of Customs and Patent Appeals in In re Langer¹ stated the following:

As a matter of Patent Office practice, a specification which contains a disclosure of utility which corresponds in scope to the subject matter sought to be patented must be taken as sufficient to satisfy the utility requirement of § 101 for the entire claimed subject matter unless there is a reason for one skilled in the art to question the objective truth of the statement of utility or its scope. (emphasis in original)

Since the Examiner has recognized a credible utility exists for both claims 6 and 17, a credible utility must also be recognized for the subject matter recited in claims 1 and 12. Therefore, the utility requirement of 35 U.S.C. § 101 (i.e., whether the invention produces a useful, concrete, and tangible result) has been met.

**THE REJECTION OF CLAIMS 1-4, 8-9, 12-15, AND 19-20 UNDER 35 U.S.C. § 102 FOR
ANTICIPATION BASED UPON ALEXANDER**

For convenience of the Honorable Board in addressing the rejections, and independent claim 12 stands or falls together with independent claim 1; claim 13 stands or falls together with claim 2; claim 14 stands or falls together with claim 3; claim 15 stands or falls together with claim 4; claim 19 stands or falls together with claim 8; and claim 20 stands or falls together with claim 9.

The factual determination of anticipation under 35 U.S.C. § 102 requires the identical disclosure, either explicitly or inherently, of each element of a claimed invention in a single reference.² As part of this analysis, the Examiner must (a) identify the elements of the claims, (b) determine the meaning of the elements in light of the specification and prosecution history,

¹ 503 F.2d 1380, 1391 USPQ 288, 297 (CCPA 1974).

² In re Rijckaert, 9 F.3d 1531, 28 USPQ2d 1955 (Fed. Cir. 1993); Lindermann Maschinenfabrik GMBH v. American Hoist & Derrick Co., 730 F.2d 1452, 221 USPQ 481 (Fed. Cir. 1984).

and (c) identify corresponding elements disclosed in the allegedly anticipating reference.³ This burden has not been met.

Claims 1 and 12

On page 4 of the Office Action, the Examiner asserted that column 6, lines 25-28 discloses the claimed routine registration means. However, upon reviewing this passage, which refers to step 172 in Fig. 7, Appellants are unable to determine where Alexander discloses that for each thread being executed in a multithreading environment, a routine is registered. Instead, Alexander only states that certain information is obtained, but there is no description of "registration" or that registration occurs for "each thread being executed," as claimed.

Appellants originally presented these arguments in the first full paragraph on page 4 of the Response. The Examiner responded to these arguments in the fourth enumerated paragraph on page 2 of the Second Office Action, in which the Examiner asserted:

4. At the top of page 4 regarding claims 1 and 12, Applicants essentially argue that the prior art, i.e. Alexander, does not disclose that a routine is not registered for each executed thread. This argument is not persuasive since Alexander discloses "a call stack tree is maintained for each thread" (column 6 lines 20-21).

Claim 1 further recites a level determination means, for which the Examiner cited column 6, lines 37-39 to disclose. This passage within Alexander, however, is silent as to (i) tracing processing being executed based upon the registration information, and (ii) a determination of a level of the tracing process. Instead, Alexander only queries whether or not a module being entered is a child or the current tree node. The "level" allegedly disclosed by Alexander regards a tree node, not the tracing process.

³ Lindermann Maschinenfabrik GMBH v. American Hoist & Derrick Co., supra.

Appellants originally presented these arguments in the second full paragraph on page 4 of the Response. The Examiner responded to these arguments in the fourth enumerated paragraph on pages 2 and 3 of the Second Office Action, in which the Examiner asserted:

Also in regard to claim 1, Applicants argue that the Alexander reference does not disclose "(i) tracing processing being executed based upon the registration information." It is noted that this language is not found in the claim, which is directed to "determining a level of a tracing process for the routine being executed based on registration information." Alexander discloses determining "the thread's 'current tree node'." In this case, the tree is interpreted as the registration information, the level is related to the node level, and the routine is the thread. Applicants further argue in regard to claim 1, that the Alexander reference does not disclose "(ii) a determination of a level of the tracing process." However, as implied above, this is related to Alexander's identification of a tree node. Therefore, Applicants' arguments are not persuasive, and the rejection is maintained.

The Examiner's first assertion is, while referring to "(i) tracing processing being executed based upon the registration information, that "this language is not found in the claim." For ease of reference, the passage in claim 1 at issue is reproduced below:

level determination means for determining a level of a tracing process for the routine being executed based on registration information registered by the routine registration means.

As is apparent from the above language, the claimed "tracing process" operates on different levels. Also, the "level determination means" determines the level based on registration information. Therefore, the tracing processing is executed based upon the registration information. Thus, the Examiner's assertion with regard to this language not being found in the claim is only correct in that the Examiner was referring to the precise language since the concept described by Appellants as not being identically disclosed by Alexander is found within claim 1.

Appellants note the Examiner's assertion that "the tree is interpreted as the registration information." This assertion by the Examiner, however, is unclear as to whether or not the

"registration information" found is the tree is intended to disclose all the registration information or just the registration information of a single node. As stated by Alexander in column 4, lines 33-34 while referring to column 4, "an event tree ... reflects call stacks observed during system execution." Alexander further describes a call stack, in column 4, lines 4-7, as "an ordered list of routines (i.e. modules, functions, methods, etc.) that have been entered, but not yet exited, at any given moment during the execution of a program." Thus, the event tree (i.e., the Examiner's alleged "registration information") of Alexander contains multiple call stacks, which include ordered lists of routines entered but not yet exited.

The Examiner further asserted that "the level is related to the node level, and the routine is the thread." At the outset, Appellants are unclear as to the Examiner's statement that "the routine is the thread" since both the claims and Alexander refer to routines and threads. Moreover, Appellants cannot rely upon the Examiner's prior comparisons in the same sentence since when the Examiner stated "the tree is interpreted as the registration information, the level is related to the node level," in the first clause, the reference to the teaching in Alexander came first, and in the second clause, the reference to the teaching in Alexander came second. Thus, the Examiner's statement the "the routine is the thread" is entirely ambiguous.

Notwithstanding the ambiguity of this statement, the Examiner's assertion that "the routine is the thread," on its face, appears to be in error. A routine is a part of a thread, but the routine is not the thread (see column 5, lines 44-46 of Alexander). Thus, for the Examiner to assert that "the routine is the thread," without further explanation, is to interpret the claim

language in a manner not in accordance with the plain and ordinary meaning attributed to that particular claim language by one having ordinary skill in the art.

As to the Examiner's assertion that "the level is related to the node level," for sake of clarity, Appellants note that the claimed "level" is "a level of a tracing process for the routine being executed" and that the tracing process is recited in claim 1 as being "for collecting information related to multithreading processing status." On the contrary, the "node level" described by Alexander is completely unrelated to a tracing process for collecting information related to multithreads processing status.

A discussion of "node level" is not found in Alexander. However, a discussion of "tree level" is described in column 5, lines 40-55. The "Level 130" is described as "the tree level (counting from the root as level 0) of the node and that "[i]ndent 140 depicts the level of the tree in an indented manner." Based upon the assumption that the second and third entries in the Level 130 column of Fig. 5 are incorrect (these entries should be "1" and "2" instead of "0" and "1"), then based upon the rest of the entries, the Level 130 essentially corresponds to the number of routines in the call stack. For example, reference is made to timestamp 12 in Fig. 3, which corresponds to feature 112 in Fig. 4 and to the second to last row in Fig. 5 of Alexander. The level of "5" in column 130 corresponds to the five indents in column 140 in Fig. 5. This "5" also corresponds to the number of routines in the call stack in Fig. 3 and to the level of the tree in Fig. 4. A similar comparison can be made, for example, between timestamp 15 in Fig. 3; feature 110 in Fig. 4, and the last row in Fig. 5.

As evident from these teachings, the tree level taught by Alexander is not a level of a tracing process for a routine being executed, as claimed. Moreover, the tree level of Alexander is not based on "registration information registered by the routine registration means." Instead, the tree level is based upon the number of routines in a call stack.

Notwithstanding all the other differences between Alexander and the claimed invention, as recited in claim 1, Appellants note that claim 1 recites "a tracing process for collecting information related to multithreading processing status." The Examiner, however, has not identified where multithreading processing status information is being collected by a "tracing process." Moreover, Appellants note that the Examiner has yet to specifically identify the specific feature in Alexander that corresponds to the claimed tracing process.

Claims 2 and 13

The Examiner asserted that column 6, lines 37-43 identically discloses the limitations recited in claim 2. Appellants respectfully disagree. Claim 2 includes the concept that level determination is based upon whether a routine identical to (or in a predetermined relation to) the routine being executed is registered. *Assuming arguendo* that the child node determination taught by step 180 of Alexander corresponds to the claimed level determination, Alexander is silent as to this level determination is based upon whether a routine (other than the routine being executed) is registered.

Appellants originally presented these arguments on page 4 of the Response. The Examiner responded to these arguments in the fifth enumerated paragraph on page 3 of the Second Office Action, in which the Examiner asserted:

5. At the bottom of page 4, Applicants essentially argue with respect to claims 2 and 13 that the Alexander reference does not disclose determining a level based on a registered routine. However, Alexander describes "if the module is already a child node." Here, the child node corresponds to the registered routine. Thus, Applicants' arguments are not persuasive and the rejection is maintained. (emphasis added)

For ease of reference, claim 2 is reproduced below:

wherein the level determination means performs the level determination of the tracing process for the routine being executed based on whether a routine identical to, or in a predetermined relation with, the routine is registered.

As already noted above, the level of the tree/node taught by Alexander is unrelated to the "level of a tracing process for the routine being executed," as claimed. Appellants also note that the Examiner's assertion that "the child node corresponds to the registered routine" is factually incorrect. A generic description by Alexander as to what constitutes a node is found in column 4, lines 33-43, and a more specific example is found in column 4, lines 44-58. As described in column 4, lines 3-4 of Alexander, "the tree reflects the call stacks observed during executing," and the node can include the following statistics: "the number of distinct times the call stack is produced, the sum of the time spent in the call stack, the total time spent in the call stack plus the time in those call stacks invoked from this call stack (referred to as cumulative time), and the number of instances of this routine above this instance (indicating depth of recursion)."

For ease of reference, the Examiner cited passage of column 6, lines 37-43 of Alexander is reproduced below:

If the trace event is an enter event, a check is made to determine if the module is already a child node of the current tree node (step 180). If not, a new node is created for the module and it is attached to the tree below the current tree node (step 182). The tree is then traversed to the module's node, and the current tree node is set equal to the module node (step 184).

Notwithstanding the claimed "level" and the "level" disclosed by Alexander are completely different, Alexander fails to teach the other portions of claim 2. Specifically, Alexander fails to teach determining the level based upon whether a routine is identical to, or in a predetermined relation with, the routine is registered. Alexander does not teach the comparison of routines. Instead, Alexander teaches determining if the module is already a child node of the current tree node. However, as already described above, Alexander fails to teach that nodes and routines are comparable. Thus, Alexander further fails to identically disclose the claimed invention, as recited in claims 2 and 13.

Claims 3 and 14

Claim 3 recites a table, which the Examiner asserted is disclosed by Fig. 3 of Alexander. However, Fig. 3 is described by Alexander as "an illustration of a trace sequence" (column 3, line 6). Thus, Fig 3 fails to identically disclose the claimed invention recited in claim 3.

Appellants originally presented these arguments on page 5 of the Response. The Examiner responded to these arguments in the sixth enumerated paragraph on page 3 of the Second Office Action, in which the Examiner asserted:

6. At the top of page 5, Applicants essentially argue with respect to claims 3 and 14 that Alexander does not disclosed a table. This argument is not persuasive since Alexander discloses a table in column 4 lines 13-15 as static information in a trace text file. Thus, the table of Fig. 3 is not simply an "illustration" as suggested by Applicants, but is interpreted at least as a table in a text file. The rejection is maintained.

Appellants further note that the table is described as being one "in which routines having a predetermined mutual relation are specified" (emphasis added). The trace sequence illustrated by Fig. 3 of Alexander does not specify routines having a predetermined mutual relation.

The trace sequence in Fig. 3 of Alexander is exactly what it purports to be: a tracking of events, as they occur, with the addition of a time stamp or an indication of the sequence in which the events occurred (see column 1, lines 36-48 and column 4, lines 12-25 of Alexander). Even if any of the events recorded by the trace have a predetermined mutual relation, this predetermined mutual relation is not captured with Fig. 3. Instead, Fig. 3 only captures the events and the order/time in which the events occurred. Thus, Alexander further fails to identically disclose the claimed invention, as recited in claims 3 and 14.

Claims 4 and 15

The Examiner cited column 4, lines 18-20 to identically disclose the limitations recited in claim 4. However, the teaching of "dynamically obtaining" information to create the trace in Fig. 3 is not comparable to the claimed means for accepting input for setting or changing the table.

Appellants originally presented these arguments on page 5 of the Response. The Examiner responded to these arguments in the seventh enumerated paragraph on pages 3 and 4 of the Second Office Action, in which the Examiner asserted:

7. Further on page 5, Applicants essentially argue in regard to claims 4 and 15 that Alexander's disclosure is not comparable to the claimed elements. Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a

patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

The Examiner has the burden of establish a prima facie case of anticipation, and as part of this burden, the Examiner is required to establish that the applied prior art identically discloses each element of the claimed invention. Claim 4 recites "means for accepting input for setting or changing the table," and as already noted, the Examiner asserted that the teaching of "dynamically obtaining" information to create the trace in Fig. 3 of Alexander discloses this particular feature. As readily apparent, the teachings of Alexander do not identically correspond to the language of the claims. Thus, the Examiner is burdened to explain why one having ordinary skill in the art would consider the teachings of Alexander identified by the Examiner to identically disclose the claimed limitations. This burden, however, has not been met. Thus, Alexander further fails to identically disclose the claimed invention, as recited in claims 4 and 15.

Claim 8 and 19

The Examiner cited column 5, lines 50-52 to identically disclose the limitations recited in claim 8. Appellants respectfully disagree. The cited passage of column 5, lines 50-52 is unrelated to the description of Fig. 7 in column 6, lines 17-47, which is the basis of the Examiner's initial description of the level determination means, which is being further described in claim 8. The Examiner has not reconciled how the calls 134 disclosed in Fig. 5 of Alexander are related to step 180 illustrated in Fig. 7. Moreover, the Examiner has failed to establish exactly what feature identically discloses the claimed "simultaneous execution information."

Appellants originally presented these arguments on page 5 of the Response. The Examiner responded to these arguments in the eighth enumerated paragraph on page 4 of the Second Office Action, in which the Examiner asserted:

8. At the bottom of page 5 through the top of page 6, Applicants essentially argue with respect to claims 8, 9, 19 and 20, that the passage cited in the Office Action is unrelated to previously cited passages, i.e. that Fig. 5 is not presented in the same context as Fig. 7, and is therefore not combinable. This argument is not persuasive. Alexander's invention is directed to storing trace information as a "tree of events" (column 2 lines 25-26). FIGS. 5 and 7 are both descriptive of this tree of events and are therefore complementary. The "simultaneous execution information" is interpreted as Alexander's "RL 132," which describes the recursion level. Therefore, the rejections are maintained. (emphasis added)

Appellants note that the underlined portion of the above-reproduced assertion by the Examiner is not factually correct. The recursion level of Alexander is described in column 4, lines 42-43 and column 5, lines 33-35 as the number of instances of this routine above this instance. Since the instances of the routine "above this instance" refers to instances of the routine being performed at a different time, the recursion level of Alexander is unrelated to adding "simultaneous execution information to registration information," as claimed.

Claims 9 and 20

The Examiner cited column 5, lines 47-48 to identically disclose the limitations recited in claim 9. Appellants incorporate herein the arguments previously presented with regard to claims 8 and 19, upon which claims 9 and 20 respectively depend. The Examiner has also not reconciled how the table data illustrated in Fig. 5 of Alexander is related to step 180 illustrated in Fig. 7.

Appellants originally presented these arguments on page 6 of the Response, and the Examiner response has already been reproduced above. As already noted, the recursion level of

Alexander is unrelated to adding "simultaneous execution information to registration information," as claimed.

Therefore, for the reasons stated above, Appellants respectfully submit that the imposed rejection of claims 1-4, 8-9, 12-15, and 19-20 under 35 U.S.C. § 102 for anticipation based upon Alexander is not viable.

THE REJECTION OF CLAIMS 5 AND 16 UNDER 35 U.S.C. § 103 FOR OBVIOUSNESS BASED UPON ALEXANDER IN VIEW OF BROBERG

For convenience of the Honorable Board in addressing the rejections, and claim 16 stands or falls together with claim 5.

Claims 5 and 16 depend ultimately from independent claims 1 and 12, and Appellants incorporates herein the arguments previously advanced in traversing the imposed rejection of claims 1 and 12 under 35 U.S.C. § 102 for anticipation based upon Alexander. The secondary reference to Broberg does not cure the argued deficiencies of Alexander. Accordingly, even if one having ordinary skill in the art were motivated to modify Alexander in view of Broberg, the proposed combination of references would not yield the claimed invention.

Moreover, Appellants note that the Examiner has failed to establish a realistic motivation to modify Alexander in view of Broberg. The Examiner's asserted motivation for this combination is "to trace such routines in order to handle various I/O activities as suggested by Broberg." The Examiner has not explained how basing a level determination on whether two routines access a

shared resource enables some unstated entity "to handle various I/O activities." Appellants, therefore, respectfully submit that the imposed rejection of claims 5 and 16 under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Broberg is not viable and, hence, solicit withdrawal thereof.

Appellants originally presented these arguments on pages 7 and 8 of the Response. The Examiner responded to these arguments in the ninth enumerated paragraph on page 4 of the Second Office Action, in which the Examiner asserted:

9. In response to applicant's argument with respect to the rejection of claims 5 and 16 at the bottom of page 6 through page 7, that the combination of Alexander and Broberg does not yield the claimed invention, the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art. See *in re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981). In this case, Broberg suggests that shared resources, such as I/O activities and others used by kernel level threads, should be traced. This provides motivation to one of ordinary skill in the art to modify Alexander according to the teaching of Broberg. Thus, this argument is not persuasive, and the rejections are maintained. (emphasis added)

The Examiner has failed to address the issue raised by Appellants. Specifically, the Examiner has not explained how basing a level determination on whether two routines access a shared resource enables some unstated entity "to handle various I/O activities." Moreover, the Examiner has not even established that Alexander in view of Broberg teaches or suggests this particular limitation (i.e., basing a level determination on whether two routines access a share resource). Instead, the Examiner has only asserted that Broberg teaches tracing shared resources. Appellants, therefore, respectfully submit that the imposed rejection of claims 5 and 16 under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Broberg is not viable.

**THE REJECTION OF CLAIMS 6-7 AND 17-18 UNDER 35 U.S.C. § 103 FOR OBVIOUSNESS
BASED UPON ALEXANDER IN VIEW OF LARUS**

For convenience of the Honorable Board in addressing the rejections, and claim 17 stands or falls together with claim 6, and claim 18 stands falls together with claim 7.

Claims 6-7 and 17-18 depend ultimately from independent claims 1 and 12, and Appellants incorporate herein the arguments previously advanced in traversing the imposed rejection of claims 1 and 12 under 35 U.S.C. § 102 for anticipation based upon Alexander. The secondary reference to Larus does not cure the argued deficiencies of Alexander. Accordingly, even if one having ordinary skill in the art were motivated to modify Alexander in view of Larus, the proposed combination of references would not yield the claimed invention.

Regarding claim 6, after reviewing page 57, column 1 of Larus, Appellants are unable to determine where "Larus teaches a level determination scheme that provides an indication of unrestricted execution upon encountering 'impossible instructions,'" as alleged by the Examiner. Appellants are unable to find any mention of a routine being provided "unrestricted execution" upon the "impossible instructions" being encountered.

Appellants originally presented these arguments on page 8 of the Response. The Examiner responded to these arguments in the tenth enumerated paragraph on page 5 of the Second Office Action, in which the Examiner asserted:

10. At the bottom of page 7 through page 8, Applicants essentially argue that the Larus reference does not teach "unrestricted execution" in response to registered information. In this case, the registered information is the sets of instructions (easy, hard, or impossible) that prompt tracing. Easy and hard instructions enjoy unrestricted execution since they do not need to be

traced. See the bottom of column 1 on page 57: "This technique only requires a tracing system to record results from impossible instructions." The rejection is maintained.

Upon reviewing these comments by Examiner, Appellants are unable to determine what is the ultimate conclusion of the Examiner's arguments. Notwithstanding the dubious comparisons between the claimed limitations and certain of the teachings identified in Larus, the Examiner has still failed to establish how and where "Larus teaches a level determination scheme that provides an indication of unrestricted execution upon encountering 'impossible instructions'," as alleged by the Examiner. The teaching by Larus that "[t]his technique only requires a tracing system to record results from impossible instructions" is unrelated to indicating permission for unrestricted execution of a tracing process if a routine is identical to, or in a predetermined relation with, the routine being executed is registered.

As to claim 7, which recites that the level determination means block execution of a tracing process unless a routine identical to the routine (or having a predetermined relation to the routine) is registered, the Examiner's comments do not support the assertion that Larus discloses this limitation. Even assuming that Larus teaches that certain types of instructions do not need to be traced unless they are registered, the claimed invention is directed to determining whether or not a first routine is blocked from being traced based upon whether or not a second routine, which has some relationship with the first routine, is registered. The Examiner's comments are only directed to a single routine, and not to the relationship between two different routines.

Appellants originally presented these arguments on page 8 of the Response. The Examiner responded to these arguments in the tenth enumerated paragraph on page 5 of the Second Office Action, in which the Examiner asserted:

Applicants further argue with respect to claim 7, that Laws does not teach blocking based upon a second routine. This argument is not persuasive since the plain language of the claim does not necessarily call for a second routine. The claim simply calls for an "identical" routine along with alternative language ("or in a predetermined relation") that might call for a second routine. Use of the alternative language allows interpretation of the claim using the identical routine only. The rejection is maintained.

The claimed language of "if a routine identical to ... the routine being executed is registered" describes two routines (albeit identical routines), not a single routine, as asserted by the Examiner. To interpret the term in the manner suggested by the Examiner would render this language establishing a condition meaningless since the routine being executed has already been registered, and thus, the condition has already been met, by definition. Appellants, therefore, respectfully submit that the imposed rejection of claims 6-7 and 17-18 under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Larus is not viable.

**THE REJECTION OF CLAIMS 10 AND 21 UNDER 35 U.S.C. § 103 FOR OBVIOUSNESS
BASED UPON ALEXANDER IN VIEW OF GDB**

For convenience of the Honorable Board in addressing the rejections, and claims 10 and 21 stand or fall together with independent claim 1.

Claims 10 and 21 depend ultimately from independent claims 1 and 12, and Appellants incorporates herein the arguments previously advanced in traversing the imposed rejection of claims 1 and 12 under 35 U.S.C. § 102 for anticipation based upon Alexander. The secondary reference to GDB does not cure the argued deficiencies of Alexander. Accordingly, even if one having ordinary skill in the art were motivated to modify Alexander in view of GDB, the proposed combination of references would not yield the claimed invention. Appellants, therefore, respectfully submit that the imposed rejection of claims 10 and 21 under 35 U.S.C. § 103 for obviousness based upon Alexander in view of GDB is not viable

**THE REJECTION OF CLAIMS 11 AND 22 UNDER 35 U.S.C. § 103 FOR OBVIOUSNESS
BASED UPON ALEXANDER IN VIEW OF SAUVAGE**

For convenience of the Honorable Board in addressing the rejections, and claims 11 and 22 stand or fall together with independent claim 1.

Claims 11 and 22 depend ultimately from independent claims 1 and 12, and Appellants incorporates herein the arguments previously advanced in traversing the imposed rejection of claims 1 and 12 under 35 U.S.C. § 102 for anticipation based upon Alexander. The secondary reference to Sauvage does not cure the argued deficiencies of Alexander. Accordingly, even if one having ordinary skill in the art were motivated to modify Alexander in view of Sauvage, the proposed combination of references would not yield the claimed invention. Appellants, therefore, respectfully submit that the imposed rejection of claims 11 and 22 under 35 U.S.C. § 103 for obviousness based upon Alexander in view of Sauvage is not viable.

Conclusion

Based upon the foregoing, Appellants respectfully submit that the Examiner's rejections under 35 U.S.C. §§ 101, 102, 103 are not viable. Appellants, therefore, respectfully solicit the Honorable Board to reverse the Examiner's rejections under 35 U.S.C. §§ 101, 102, 103.

To the extent necessary, a petition for an extension of time under 37 C.F.R. § 1.136 is hereby made. Please charge any shortage in fees due under 37 C.F.R. §§ 1.17, 41.20, and in

Application No.: 10/715,769

connection with the filing of this paper, including extension of time fees, to Deposit Account 09-0461, and please credit any excess fees to such deposit account.

Date: February 21, 2007

Respectfully submitted,

/Scott D. Paul/

Scott D. Paul

Registration No. 42,984

Steven M. Greenberg

Registration No. 44,725

CUSTOMER NUMBER 46320

VIII. CLAIMS APPENDIX

1. An information processor suitable for executing a routine including a tracing process for collecting information related to multithreading processing status, the information processor comprising:

routine registration means for registering a routine being executed for each thread being activated; and

level determination means for determining a level of a tracing process for the routine being executed based on registration information registered by the routine registration means.

2. The information processor according to claim 1, wherein the level determination means performs the level determination of the tracing process for the routine being executed based on whether a routine identical to, or in a predetermined relation with, the routine is registered.

3. The information processor according to claim 2, further comprising a table in which routines having a predetermined mutual relation are specified.

4. The information processor according to claim 3, further comprising means for accepting input for setting or changing the table.

5. The information processor according to claim 2, wherein the routines in a predetermined relation are routines that access a shared resource.

6. The information processor according to claim 1, wherein the level determination means indicates permission, in determining the level of the tracing process for the routine being executed, for unrestricted execution of a tracing process if a routine identical to, or in a predetermined relation with, the routine being executed is registered.

7. The information processor according to claim 1, wherein the level determination means blocks, in determining the level of the tracing process for the routine being executed, execution of a tracing process unless a routine identical to, or in a predetermined relation with, the routine being executed is registered.

8. The information processor according to claim 1, wherein, responsive to registration of a routine, the level determination means adds, if there is already registered a routine identical to or in a predetermined relation with the routine being registered, simultaneous execution information to registration information of both the routine being registered and the routine already registered.

9. The information processor according to claim 8, wherein the simultaneous execution information includes trace level information indicating the level of the tracing process.

10. The information processor according to claim 1, further comprising a server program including the routine and the tracing process;

an application execution control part for executing the routine in response to a call of the routine by a client program; and

a trace executioner for executing the tracing process in response to a request by the application execution control part;

wherein the trace executioner queries, when executing the tracing process, the level determination means to determine the level of the tracing process.

11. The information processor according to claim 1, comprising a program for using the routine including a tracing process;

a shared library in which the routine including the tracing process is stored; and

a trace executioner for executing the tracing process in response to a request by the routine used by the program;

wherein the trace executioner inquires, when executing the tracing process, the level determination means to determine the level of the tracing process.

12. A tracing process method in an information processor suitable for executing a routine including a tracing process for collecting information related to multithreading processing status;

the tracing process method comprising:

a routine registration step for registering a routine being executed for each thread being activated; and

a level determination step for determining a level of the tracing process for the routine being executed based on information included in the registration.

13. The tracing process method according to claim 12, wherein the level determination step determines the level of the tracing process for the routine being executed based on whether a routine identical to, or in a predetermined relation with, the routine is already registered.

14. The tracing process method according to claim 13, wherein a table, in which routines having a predetermined mutual relation are specified, is referred to in determining the level of the tracing.

15. The tracing process method according to claim 14, further comprising a step of accepting input for setting or changing the table.

16. The tracing process method according to claim 13, wherein routines in the predetermined mutual relation are routines that access a shared resource.

17. The tracing process method according to claim 12, wherein the level determination step indicates permission, in determining the level of a tracing process for the routine being executed, for unrestricted execution of the tracing process if any routine identical to, or in a predetermined relation with, the routine being executed is registered.

18. The tracing process method according to claim 12, wherein the level determination step blocks, in determining the level of a tracing process for the routine being executed, execution of a tracing process unless a routine identical to, or in a predetermined relation with, the routine being executed is registered.

19. The tracing process method according to claim 12, wherein, responsive to registration of a routine, the level determination step adds, if there is already registered a routine identical to or in a predetermined relation with the routine being registered, simultaneous execution information to registration information of both the routine already registered and the routine being registered.

20. The tracing process method according to claim 19, wherein the simultaneous execution information includes trace level information indicating the level of the tracing process.

21. The tracing process method according to claim 12, further comprising a step of executing a server program including the routine and the tracing process;

an application execution control step of executing the routine in response to a call by a client program; and

a trace execution step of executing the tracing process occurred at the application execution control step;

wherein the tracing execution step refers to the result of the determination at the level determination step in executing the tracing process.

22. The tracing process method according to claim 12, comprising a step of executing a program using the routine and the tracing process;

a step of the information processor executing the routine including the tracing process stored in a shared library; and

a tracing execution step of the information processor executing the tracing process at the routine execution step;

wherein the tracing execution step refers to the result of the determination at the level determination step in executing the tracing process.

IX. EVIDENCE APPENDIX

No evidence submitted pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132 of this title or of any other evidence entered by the Examiner has been relied upon by Appellants in this Appeal, and thus no evidence is attached hereto.

X. RELATED PROCEEDINGS APPENDIX

Since Appellants are unaware of any related appeals and interferences, no decision rendered by a court or the Board is attached hereto.